UNIVERSIDAD
DE GRANADA

Clasificación con Conjuntos de Datos No Balanceados
Laboratorio de Programación en R

Minería de Datos, Aspectos Avanzados

Dpto. Ciencias de la Computación e Inteligencia Artificial
E.T.S. de Ingenierías Informática y de Telecomunicación
Universidad de Granada

Índice

1. Introducción

En esta práctica se pretende que el estudiante comprenda las implicaciones que tiene un conjunto con clases desequilibradas (*imbalanced*) en el rendimiento de los clasificadores estándar.

La práctica se divide en tres partes bien diferenciadas:

1. Una primera parte donde se analizará el rendimiento alcanzado por los clasificadores mediante el uso de técnicas básicas de preprocesamiento. Para ello se utilizarán conjuntos de datos sencillos mediante el software R. Este estudio se realizará durante la misma clase de prácticas.
2. Una segunda parte, también a realizar durante la sesión de prácticas, en la que se analizará el comportamiento de los algoritmos de SMOTE extendidos en comparación con el SMOTE clásico.
3. Por último, y con carácter opcional se propone realizar un análisis relativamente exhaustivo de los parámetros del algoritmo SMOTE (número de vecinos, porcentaje de oversampling, enfoque de interpolación, etc.) para chequear su comportamiento en función de distintos valores.

Nota Importante: A lo largo de todo este guion de prácticas, las indicaciones se realizan para la solución implementada bajo R. Si decide guiarse por Python, tendrá que trabajar con el paquete "*imbalanced-learn*" de manera totalmente autónoma.

También es importante hacer notar que el guion de prácticas NO está diseñado como un tutorial "paso a paso", si no que se da por sentado la capacidad del estudiante para llevar a cabo las distintas tareas propuestas sin especificar todo el contenido.

2. Objetivos

Para esta sesión de laboratorio, se necesitará descargar desde PRADO2 el fichero `imbalanced.R`, además de los conjuntos de datos `subclus.txt` y `circle.txt`.

Para la evaluación, se deberán enviar a la actividad correspondiente de PRADO2 todos los ficheros R o Python que se hayan utilizado. Hay que asegurarse de que el código entregado esté suficientemente comentado.

También, será necesario proporcionar un análisis breve de los resultados obtenidos usando comentarios en los ficheros R o Python. Particularmente se valorará el uso de herramientas de generación de informes tipo KnitR.

La fecha de entrega para las diferentes actividades se indicará una vez concluya el presente curso. Es importante hacer notar la "optatividad" de esta práctica, toda vez que el grueso de la evaluación final se llevará a cabo mediante el trabajo sobre Deep Learning.

3. Análisis del efecto del desbalanceo en problemas de clasificación

En este ejercicio, se analizarán las estrategias a nivel de datos ROS, RUS y SMOTE para tratar con distribuciones de datos no balanceadas.

3.1 Preparación de datos y rendimiento básico de clasificación

Abra el fichero imbalanced.R. El código carga el conjunto de datos bidimensional subclus y lo visualiza. Calcula la ratio de imbalanceo (IR) y configura las particiones, respetando el desbalanceo de clase en cada partición.

Finalmente, evalúe el rendimiento del clasificador base kNN sobre este conjunto de datos por medio de las diferentes métricas de calidad.

Hágalo también sobre el conjunto de datos circle.

3.2 Random Oversampling (ROS)

Estudie el comportamiento de la técnica de random oversampling (ROS) en el fichero imbalanced.R.

La aplicación de ROS debería obtener un conjunto de entrenamiento perfectamente equilibrado, duplicando aleatoriamente las instancias de la clase minoritaria.

Evalúe su rendimiento sobre el conjunto de datos subclus utilizando el clasificador kNN. Hágalo también sobre el conjunto de datos circle.

3.3 Random Undersampling (RUS)

Estudie el comportamiento de la técnica de random undersampling (RUS) en el fichero imbalanced.R.

La aplicación de RUS debería obtener un conjunto de entrenamiento perfectamente equilibrado, borrando instancias de la clase mayoritaria de forma aleatoria.

Evalúe su rendimiento sobre el conjunto de datos subclus utilizando el clasificador kNN. Hágalo también sobre el conjunto de datos circle.

3.4 Synthetic Minority Oversampling Technique (RUS)

Estudie el comportamiento de la técnica de SMOTE en el fichero imbalanced.R.

La aplicación de SMOTE debería obtener un conjunto de entrenamiento perfectamente equilibrado, creando nuevas instancias de la clase minoritaria.

Evalúe su rendimiento sobre el conjunto de datos subclus utilizando el clasificador kNN. Hágalo también sobre el conjunto de datos circle.

4. Paquete imbalance y combinación de técnicas

Existe un paquete en CRAN llamado 'imbalance' que implementa algunas de las técnicas más conocidas de preprocesamiento de datos para clasificación no balanceada. La documentación se puede encontrar en

https://github.com/ncordon/imbalance.

Utilizando el paquete 'imbalance', se pide utilizar técnicas avanzadas basadas en SMOTE. Para ello, usa la función oversample:

```
oversample(dataset, ratio = NA, method = c("RACOG",
"wRACOG", "PDFOS", "RWO", "ADASYN", "ANSMOTE", "SMOTE",
"MWMOTE", "BLSMOTE", "DBSMOTE", "SLMOTE", "RSLSMOTE"),
filtering = FALSE, classAttr = "Class", wrapper = c("KNN",
"C5.0"), ...)
```

donde

```
dataset:  A binary class data.frame to balance.
ratio:    Number between 0 and 1 indicating the desired
     ratio between minority examples and majority ones,
     that is, the quotient size of minority class/size of
     majority class. There are methods, such as ADASYN or
     wRACOG to which this parameter does not apply.
method:   A character corresponding to method to apply.
     Possible methods are: RACOG, wRACOG, PDFOS, RWO,
     ADASYN, ANSMOTE, BLSMOTE, DBSMOTE, BLSMOTE, DBSMOTE,
     SLMOTE, RSLSMOTE
filtering:     Logical (TRUE or FALSE) indicating whether
     to apply filtering of oversampled instances with
     neater algorithm.
classAttr:     character Indicates the class attribute from
     dataset. Must exist in it.
wrapper:  A character corresponding to wrapper to apply if
     selected method is wracog. Possibilities are: "C5.0"
     and "KNN".
```

Verificar su rendimiento sobre el conjunto de datos subclus y circle, así los propios conjuntos de datos proporcionados por el paquete. Proporcionar visualización.

5. Análisis de los parámetros de SMOTE *(parte optativa)*

En esta última parte de la práctica se pretende analizar la influencia de los distintos parámetros de SMOTE. En efecto, durante las clases teóricas, se indicó que muchos de los parámetros de SMOTE podían tener cierta importancia de cara a la calidad de los nuevos ejemplos sintéticos generados sobre el conjunto de entrenamiento.

Por ello, el objetivo de esta tarea es contrastar algunos de ellos para comprobar cuáles pueden tener más influencia, o cuáles pueden ser valores significativos para observar diferencias en los resultados.

Para llevar a cabo este objetivo, lo primero es determinar el marco experimental, que queda a disposición del alumno.

En cuanto a conjuntos de datos a utilizar, en principio se pueden seleccionar por defecto “subclus” y “circle”, si bien el estudio será más relevante cuando mayor sea el número de problemas, tanto sintéticos como reales. En cualquier caso, deberá utilizar una técnica de validación cruzada para chequear los resultados.

Como clasificador base para analizar el comportamiento, se puede utilizar por defecto kNN con K = 1 ó K = 3. También podría ser interesante analizar los resultados con el árbol de decisión C4.5 o incluso Random Forest o cualquier otra técnica de calidad que considere apropiada.

Por último, quedaría por discutir qué parámetros son apropiados para el estudio, y qué rango de valores utilizar. Los parámetros más directos serían K para el número de vecinos escogidos (por ejemplo, entre K = 1, K = 5, K = N/2 con N número de instancias positivas, etc.), y el porcentaje de oversampling (duplicar la clase minoritaria, 50-50 de ratio de clases, un 50% de clase minoritaria sobre mayoritaria, etc.).

Para ello, puede bien realizar una implementación ad hoc de SMOTE, o analizar entre las que hay disponibles en los distintos paquetes de R, aquélla que le permita realizar un “Racing” de los parámetros.

Construya las tablas correspondientes de resultados y haga un breve análisis si observa algún patrón interesante en ellas.